A

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/544,492	04/07/2000	Swain W. Porter	004814.P014	1773

25943 7590 11/19/2003

SCHWABE, WILLIAMSON & WYATT, P.C.
PACWEST CENTER, SUITES 1600-1900
1211 SW FIFTH AVENUE
PORTLAND, OR 97204

EXAMINER	
SHAH, NILESH R	
ART UNIT	PAPER NUMBER
2127	6

DATE MAILED: 11/19/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/544,492	**Applicant(s)** PORTER, SWAIN W.	
	Examiner Nilesh R Shah	**Art Unit** 2127	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)□ Responsive to communication(s) filed on *08/23/03* .

2a)☒ This action is **FINAL.** 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)□ Claim(s) *1-16 and 19-26* is/are rejected.

7)□ Claim(s) *17 and 18* is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *07 April 2000* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)□ The proposed drawing correction filed on _____ is: a)□ approved b)□ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)□ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____ .

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)□ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) □ The translation of the foreign language provisional application has been received.

15)□ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .

4) □ Interview Summary (PTO-413) Paper No(s). _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: .

DETAILED ACTION

This action is in response to request for reconsideration filed on 8/25/03.

Response to Remarks

As per claim 1, Applicant states Christie et al (6,154,818) does not teach remapping a stored task privilege level. Examiner respectfully disagrees. Christie teaches a 'remapper is a circuit that maps an MSR address to a local address of an implemented MSR' (col 14 lines 30-65). Christie continues to teach that MSR relates to privileged levels. 'The x86 microprocessor architecture includes a plurality of architectural registers. Architectural registers are registers that are inherent to a microprocessor design. For example, in the x86 architecture eight general registers and six segment registers are defined. Each implementation of an x86 microprocessor includes these architectural registers. In addition to the architectural registers, the x86 architecture defines a set of model specific registers (MSRs). The MSRs are used to define control and status registers that may differ between various implementations of an x86 microprocessor. For example, a Time Stamp Counter is not defined as part of the x86 architecture. Some implementations of an x86 microprocessor, however, include a Time Stamp Counter as an MSR.') (col. 1 line 30 –col. 2 line 48). Christie clearly teaches the use of a privilege remapping.

As per claim 13, Applicant states Parmar et al (3,916,385) does not teach a remapper of privileged especially from a high to low level. Parmar teaches 'The 4 rings or privilege levels are identified by integers 0-3; each ring represents a level of privilege in the system with level 0 having the most privilege and level 3 the least. Level 0 is known as the inner ring and level 3 as

the outer ring. The basic notion as previously discussed is that a procedure belonging to an inner ring has free access to data in an outer ring. Conversely a procedure in an outer ring cannot access data in an inner ring without incurring a protection violation exception. Transfer of control among procedures is monitored by a protection mechanism such that a procedure execution in an outer ring cannot directly branch to a procedure in an inner ring. This type of control transfer is possible only by execution of a special "procedure-call" instruction. This instruction is protected against misuse in a number of ways.' (fig 2, and col. 9 line 45 – col. 11 line 50) In addition Parmar teaches that one ring may branch to another ring in order to change the privileges of the tasks. Finally, Parmar teaches many different rules of the system. One of the rule include an inward and outward remapping of procedure's privilege in figure 2 element 202 and 203 show that it is legal to not only move inward but outward in the ring. ('A procedure in an inner ring such as ring 2 on FIG. 2 has free access to data in an outer ring such as ring 3 and a legal access (arrow 201) results. Conversely a procedure in an outer ring such as ring 3 cannot access data in an inner ring such as ring 2 and an attempt to do so results in an illegal access (arrow 202). (24) 2. A procedure in an outer ring such as ring 3 can branch to an inner ring such as ring 1 via gate 204 which results in a legal branch 203, but a procedure operating in an inner ring such as ring 2 may not branch to an outer ring such as ring 3.
(25) 3. Each segment containing data is assigned 2 ring values, one for read (RD) and one for write (WR). These ring values specify the maximum ring value in which a procedure may execute when accessing the data in either the read or write mode.') Clearly Parmar teaches the use of a remapper privileged including a remapper that can move from high to low levels of the ring.

Claim Objections

Claims 17 and 18 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claims 12 and 20 are objected to under 37 CFR 1.75(c) as being in improper form because a multiple dependent claim. Claim 12 is dependent on claim 1 but should be dependent on method claim 10. Claim 20 is dependent on claim 21 but should be dependent on claim 19. See MPEP § 608.01(n).

Claim Rejections - 35 USC § 102

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-12 and 23-26 are rejected under 35 U.S.C. 102(b) as being anticipated by Christie (6,154,818).

As per claim 1, Christie teaches a processor comprising:

a control register to store a task privilege level for a task; (col. 1 line 56 – col. 2 line 48) and a privilege remapper coupled to the control register to dynamically remap the stored task privilege level (e.g. Fig. 3B and col. 14 lines 30 -65).

As per claim 2, Christie teaches a privilege remapper comprises a register to store a plurality of remapped task privilege levels to be accessed using the stored task privilege level prior to runtime privilege checking (e.g. Fig. 3B and col. 14 lines 30 -65).

As per claim 3, Christie teaches a privilege remapper comprises a storage array to store a plurality of set of remapped task privilege levels to be accessed using a configuration value and the stored task privilege level prior to runtime privilege checking (e.g. Fig. 3B and col. 14 lines 30 -65).

As per claim 4, Christie teaches a privilege remapper comprises one or more logical elements to logically alter one or more bits of the stored privilege level prior to runtime privilege checking (e.g. Fig. 3B, col. 5 line 1 –col. 7 line 25 and col. 14 lines 30 -65).

As per claim 5, Christie teaches a privilege remapper further comprises at least one selector coupled to at least one of the one or more logical elements to effectuate conditional performance of said logically alteration for at least one bit of the stored privilege level prior to runtime privilege checking (e.g. Fig. 3B, col. 1 line 56 – col. 2 line 48 and col. 14 lines 30 -65).

As per claim 6, Christie teaches a processor further comprises at least one selector coupled to the control register and the privilege remapper to effectuate conditional performance of said remapping of the stored task privilege level prior to runtime privilege checking (e.g. col. 3 lines 45-67, col. 4 lines 45-65, col. 6 lines 10-41).

As per claim 7, Christie teaches a method comprising:
storing a first task privilege level for a task(col. 1 line 56 – col. 2 line 48); and
dynamically remapping the first task privilege level to a second task privilege level prior to runtime privilege checking to effectuate a different execution privilege level for the task (e.g. col. 3 lines 45-67, col. 4 lines 45-65).

As per claim 8, Christie teaches a dynamic remapping comprises accessing a register to retrieve a selected one of a plurality of remapped task privilege levels stored in said register, using the stored first task privilege level, prior to runtime privilege checking (e.g. col. 3 lines 45-67, col. 4 lines 45-65)

As per claim 9, Christie teaches a dynamic remapping comprises accessing a storage array to retrieve a selected one of a plurality of remapped task privilege levels stored in said storage array in a set-wise manner, using a configuration value and the stored first task privilege level, prior to runtime privilege checking (e.g. col. 3 lines 45-67, col. 4 lines 45-65).

As per claim 10, Christie teaches a dynamic remapping comprises logically altering one or more bits of the stored first task privilege level, prior to runtime privilege checking (col.5 line 1 – col. 7 line 25).

As per claim 11, Christie teaches altering being conditionally performed (e.g. col. 3 lines 45-67, col. 4 lines 45-65, col. 6 lines 10-41).

As per claim 12, Christie teaches a dynamic remapping being conditionally performed (e.g. col. 3 lines 45-67, col. 4 lines 45-65, col. 6 lines 10-41).

As per claim 23, Christie teaches the use of a processor comprising:

a control register to store a privilege level(col. 1 line 56 – col. 2 line 48); and

a privilege remapper coupled to the control register to dynamically remap the stored privilege level prior to runtime privilege checking (e.g. col. 3 lines 45-67, col. 4 lines 45-65, col. 6 lines 10-41) .

As per claim 24, Christie teaches the use of an processor further comprises at leastone selector coupled to the control register and the privilege remapper to effectuate conditional performance of said remapping of the stored privilege level prior to runtime privilege checking.

As per claim 25, Christie teaches the use of an

a control register to store a privilege level (e.g. col. 1 line 56 – col. 2 line 48); and

a privilege remapper coupled to the control register to dynamically remap the stored privilege level prior to runtime privilege checking. (e.g. Fig. 3B and col. 14 lines 30 -65).

As per claim 26, Christie teaches one selector coupled to the control register and the privilege remapper to effectuate conditional performance of said remapping of the stored privilege level prior to runtime privilege checking (e.g. col. 1 line 56 – col. 2 line 48, Fig. 3B and col. 14 lines 30 -65).

Claims 13, 15, 16, 19-22 are rejected under 35 U.S.C. 102(b) as being anticipated by Parmar et al (hereinafter Parmar) (3,916,385).

As per claim 13, Parmar teaches a processor having a 4-ring privilege protection scheme, where tasks attributed with a lower ring privilege level is more privileged than tasks attributed with a higher ring privilege level, a method comprising (e.g. fig 2, and col. 9 line 45 – col. 11 line 50):

attributing a ring-2 privilege level to a first task, nominally giving said first task more privilege than a second plurality of tasks which are attributed with a ring-3 privilege (e.g. fig 2, and col. 9 line 45 – col. 11 line 50) and;

dynamically remapping each ring-2 privilege level to a ring-3 privilege level, and each ring-3 privilege level to a ring-2 privilege level prior to runtime privilege checking to cause said first task to execute in fact with less privileges than said second plurality of tasks(e.g. fig 2, and col. 9 line 45 – col. 11 line 50). Parmar teaches that by using element 204 and 203 it is possible to branch in or out (remap) of rings (e.g. col. 11 lines 20-50).

As per claim 15, Parmar teaches second plurality of tasks are associated with an operating system (col. 11 lines 3-17).

As per claim 16, Parmar teaches a method comprising:

attributing a first privilege level to a first collection of programming instructions, said first privilege level being different from a second privilege level assigned to a second collection of programming instructions, resulting in said first collection of programming instructions to execute with a first relative privilege relationship to said second collection of programming instructions at execution time(e.g. fig 2, and col. 9 line 45 – col. 11 line 50).; and

dynamically remapping said first privilege level to a third privilege level prior to runtime privilege checking to cause the first collection of programming instructions to execute with a second different relative privilege relationship to said second collection of programming instructions (e.g. fig 2, and col. 9 line 45 – col. 11 line 50). Parmar teaches that by using element 204 and 203 it is possible to branch in or out (remap) of rings (e.g. col. 11 lines 20-50).

As per claim 19, Parmar teaches a method comprising:
attributing a first more privileged privilege level to a first subset of least privileged tasks attributed with a least privileged privilege level (e.g. fig 2, and col. 9 line 45 – col. 11 line 50); and
dynamically remapping said first more privileged privilege level attributed to said first subset of least privileged tasks to said least privileged privilege level, and remapping said least privileged privilege level attributed to residual ones of said least privileged tasks prior to runtime privilege checking to cause said first subset of least privileged tasks to execute with lesser privileges than said residual ones of the least privileged tasks (e.g. fig 2, and col. 9 line 45 – col. 11 line 50). Parmar teaches that by using element 204 and 203 it is possible to branch in or out (remap) of rings (e.g. col. 11 lines 20-50).

As per claim 20, Parmar teaches a least privileged privilege level of said residual ones of said least privileged tasks are remapped to said first more privileged privilege level (e.g. fig 2, and col. 9 line 45 – col. 11 line 50). Parmar teaches that by using element 204 and 203 it is possible to branch in or out of rings (e.g. col. 11 lines 20-50).

As per claim 21, Parmar teaches a method comprising:

attributing a first lesser privileged privilege level to a first subset of most privileged tasks attributed with a most privileged privilege level (e.g. fig 2, and col. 9 line 45 – col. 11 line 50); Parmar teaches that by using element 204 and 203 it is possible to branch in or out (remap) of rings (e.g. col. 11 lines 20-50) and

dynamically remapping said first lesser privileged privilege level attributed to said first subset of most privileged tasks to said most privileged privilege level, and remapping said most privileged privilege level attributed to residual ones of said most privileged tasks prior to runtime privilege checking to cause said residual ones of the most privileged tasks to execute with lesser privileges than said first subset of most privileged tasks (e.g. fig 2, and col. 9 line 45 – col. 11 line 50). Parmar teaches that by using element 204 and 203 it is possible to branch in or out (remap) of rings (e.g. col. 11 lines 20-50).

As per claim 22, Parmar teaches wherein said most privileged privilege level of said residual ones of said most privileged tasks are remapped to said first lesser privileged privilege level (e.g. fig 2, and col. 9 line 45 – col. 11 line 50). Parmar teaches that by using element 204 and 203 it is possible to branch in or out (remap) of rings (e.g. col. 11 lines 20-50).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Parmar et al (hereinafter Parmar) (3,916,385).

As per claim 14, Parmar does not specifically teaches wherein said first task is associated with an Internet application. Official notice is taken that tasks assigned to Internet application are well known. It would be obvious to one skilled in the art to use an Internet application as a first task in order to provide a source of external communications.

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nilesh R Shah whose telephone number is 703-305-8105. The examiner can normally be reached on Monday-Friday 8am-4pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Grant can be reached on 703-308-1108. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

MAJID A. BANANKHAH
PRIMARY EXAMINER

NS
November 3, 2003